Implementing Domain Engineering

Presented By

Ravinn Chhut

CACI, Inc.

(703) 679-3852

rchhut@caci.com

Course Outline

- Introduction
- Description of the Domain Engineering process
  - Domain Analysis
  - Domain Design
  - Domain Implementation
- Summary

Course Goals

- Become knowledgeable of the process
  - Understand the importance of Domain Engineering and the products generated
  - Understand the steps for conducting:
    - Domain Analysis
    - Domain Design
    - Domain Implementation

Group Profile

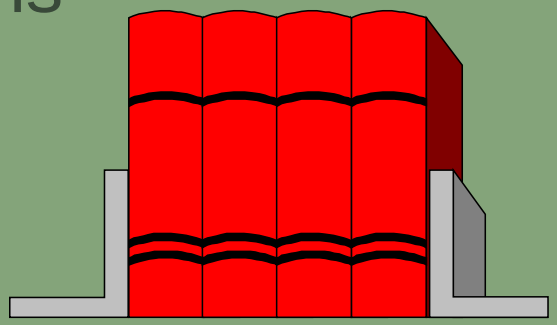
- Who are you?
- What organization do you work for?
- What project are you working on?
- Do you have any experience with domain engineering?
- What are your expectations of this class?

Domain Analysis

Identify the Domain

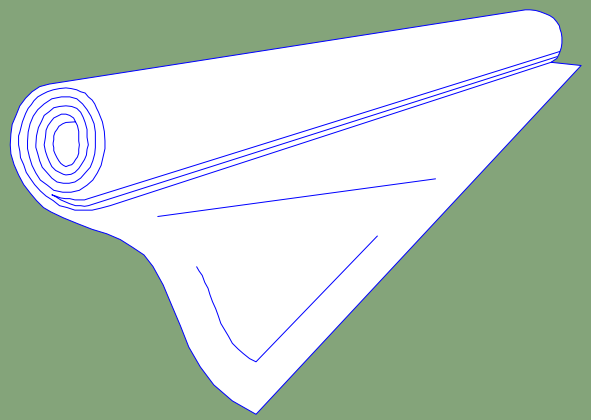
Identify Information Sources

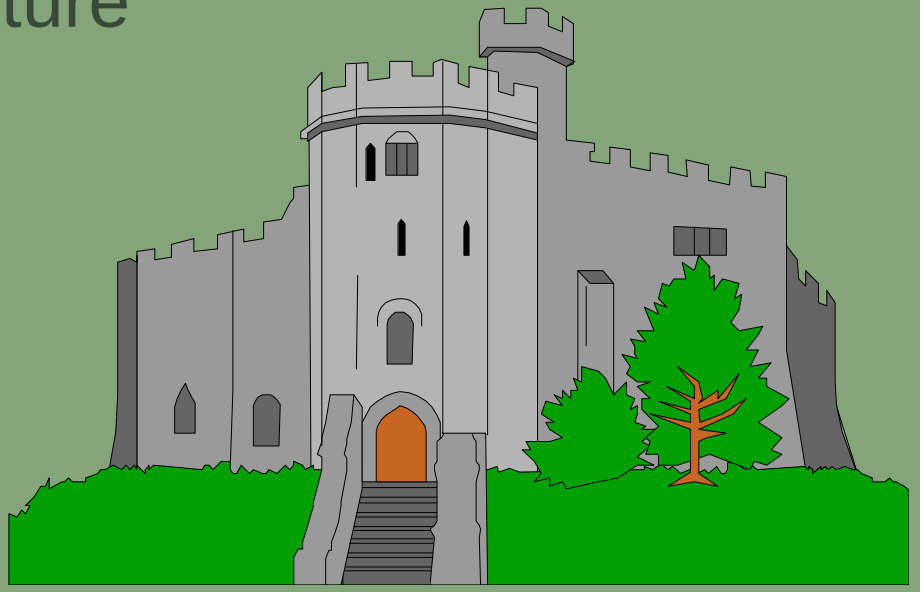
- Identify reliable domain information and expertise
  - Existing and planned systems
    - » Management personnel
    - » Technical personnel
    - » User/Operator documentation
    - » Training material
  - Functional (domain) experts



Identify Information Sources

- Domain Knowledge
  - Domain Models
  - Domain Architecture





Sources of Information

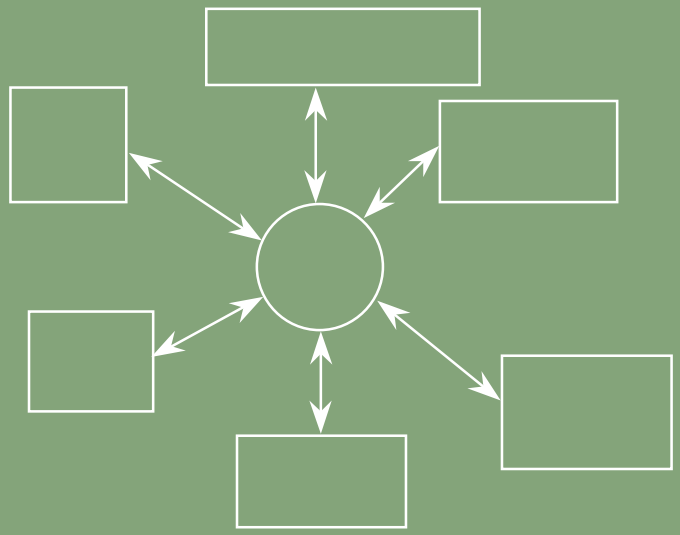
Domain Expertise	Documentation		
	MIL-STD-7935A	DOD-STD-2167A	MIL-STD-498
Functional Proponents Users System Developers Seminars & Training Trainers	Functional Description System/Sub-System Specification Software Unit Specification Database Specification Users Manual	System/Segment Design Document Software Requirements Specification Interface Requirements Specification Interface Design Document Software Design Document	Operational Concept Description System/Subsystem Specification Software Development Plan Software Requirements Specs Interface Requirements Specs Interface Design Description Software Input/Output Manual Software Users Manual Software Center Operator Manual

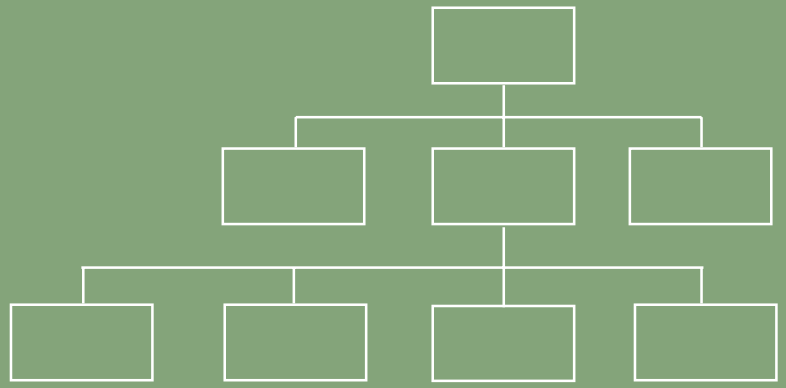
Gather Domain Information

- Using the information sources earlier identified:
  - Conduct interviews with experts
  - Review the documentation
  - Attend training sessions
  - Use existing systems
  - Do independent research

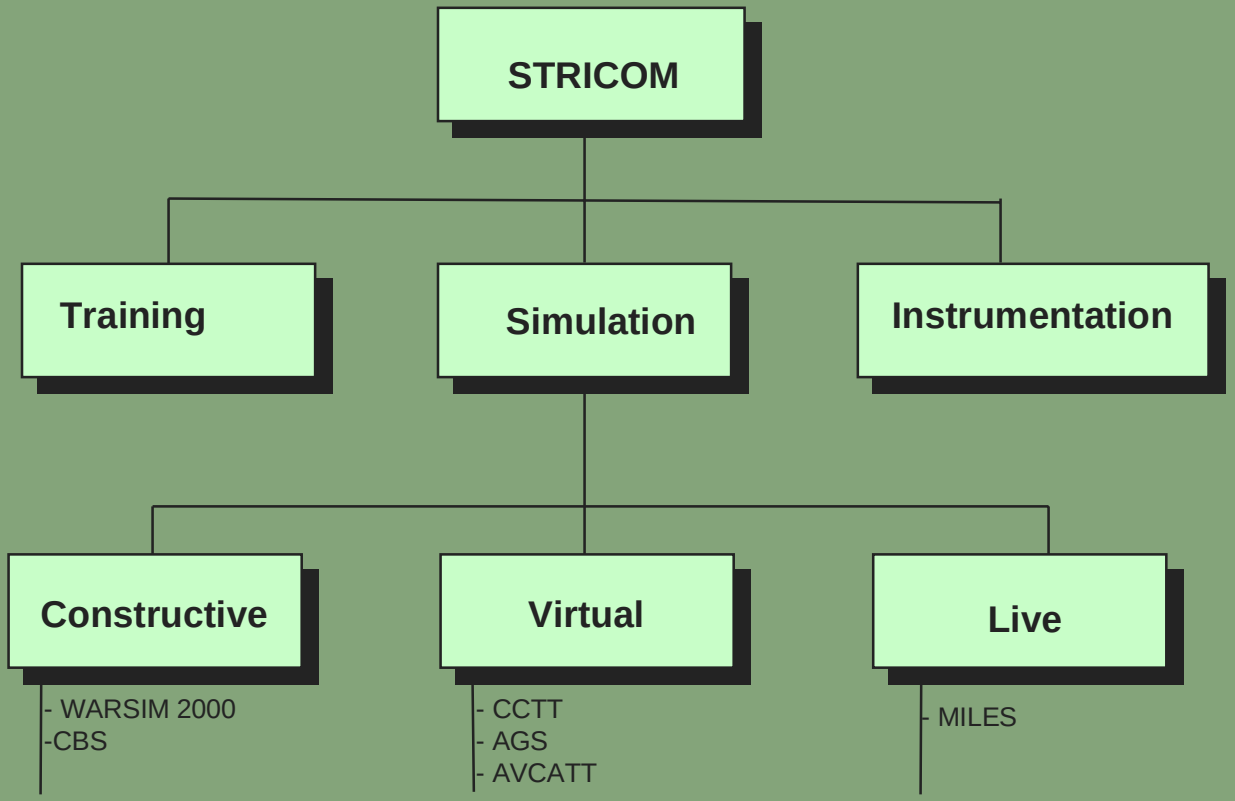
Describe the Domain

- Establish relationships with other domains
- Determine high-level functions





Describe the Domain



Note: This is just an example

Verify Domain Description

- Domain experts independently review the products for accuracy
- Any discrepancies are resolved
- Domain manager approves domain description

Scope the Domain

Establish Scoping Criteria

CRITERION	VALUE	WEIGHT
Number of Existing Systems	1-2 3-5 6-n	1 2 3
Future Systems Development	None One or more	0 1
Vertical Reuse	None 1 family 2 or more families	0 1 3
Horizontal Reuse	None or 1 family 2 families 3 or more families	0 1 3
Reuse Expertise	Low Medium High	1 2 3
Stability	Low Medium High	1 2 3
Participation	Low Medium High	1 2 3
Available Information	Low Medium High	1 2 3

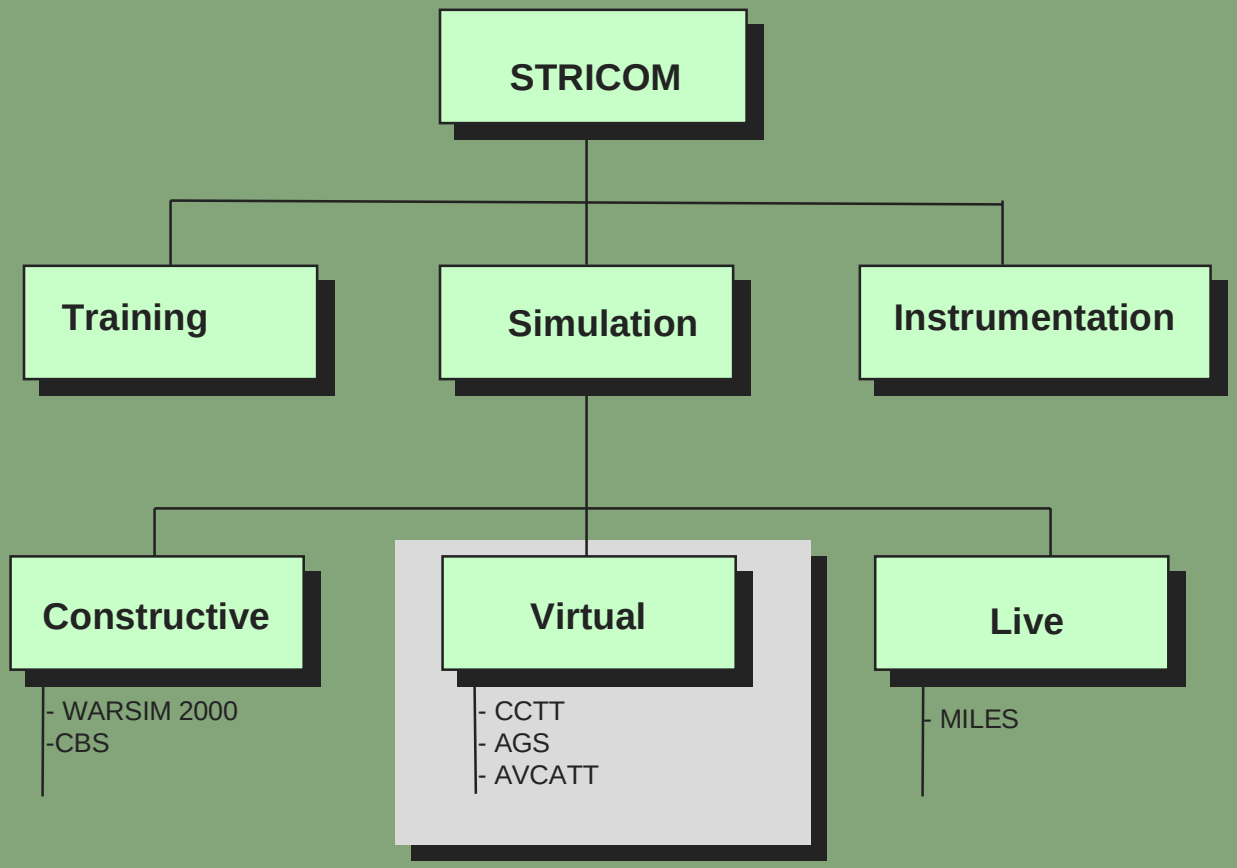
Define Domain Boundaries

Selection Criteria / Candidate Subdomains	# of Existing Systems	Future Systems Development	Vertical Reuse	Horizontal Reuse	Reuse Expertise	Stability	Participation	Available Information	Score
Training	0	1	0	0	1	2	2	2	8
Instrumentation	0	1	1	0	3	3	2	3	13
Virtual	**3**	**1**	**3**	**3**	**2**	**3**	**3**	**3**	**21**
Constructive	2	1	3	1	3	3	2	3	18
Live	1	1	1	0	1	3	3	2	12

Note: This is just an example

Verify Domain Scope

Model the Domain of Focus

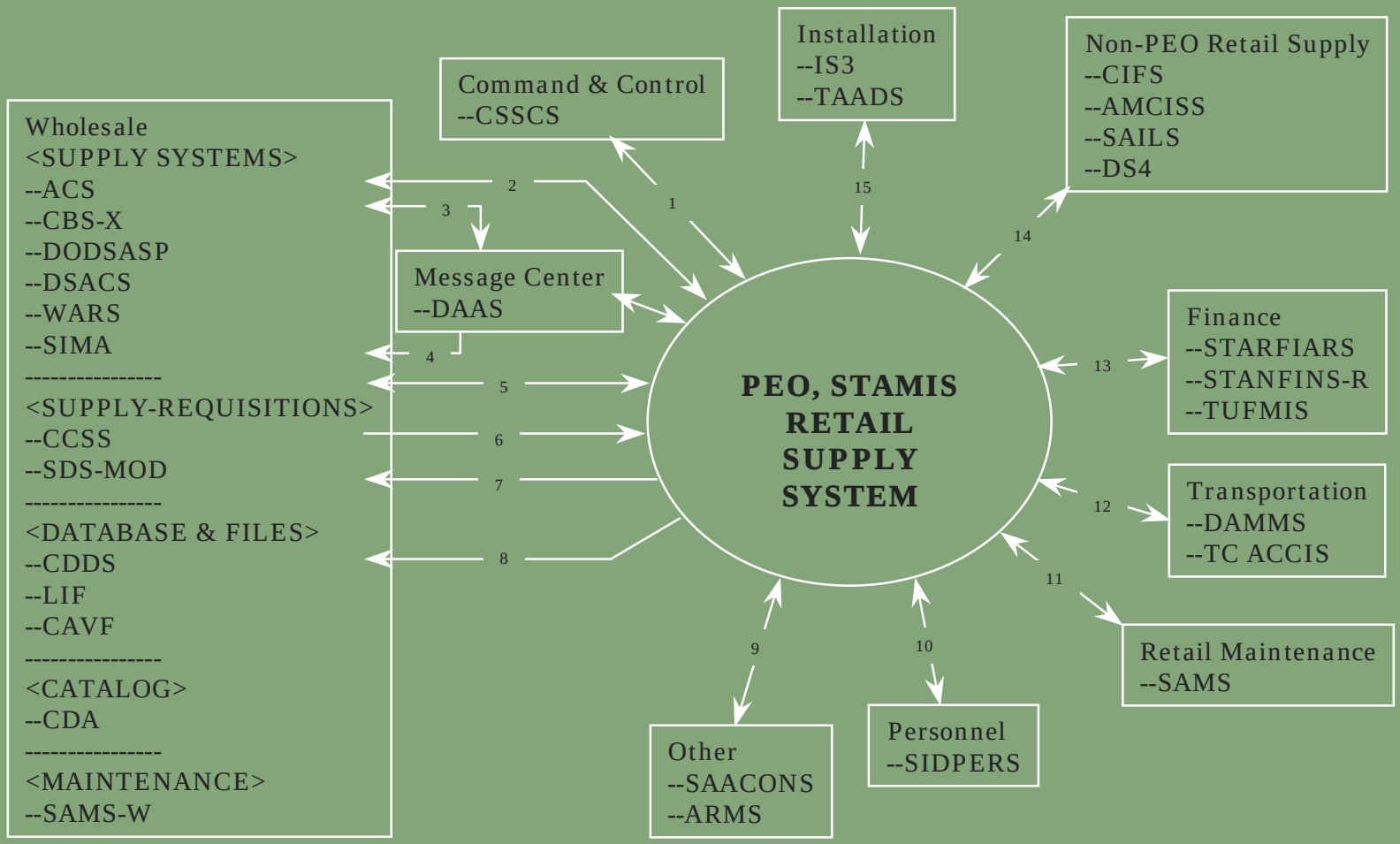


Analyze the Domain

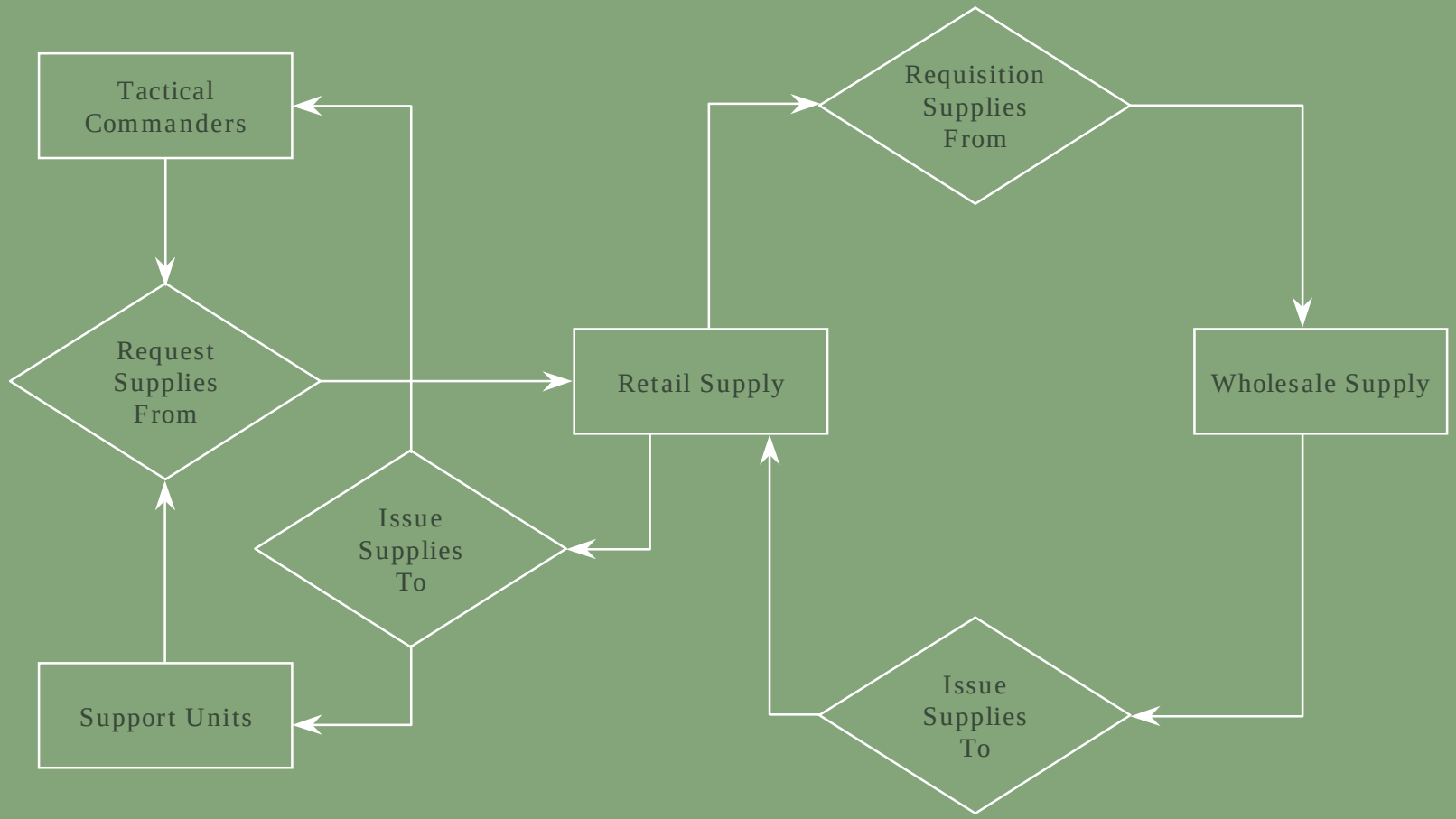
Organize & Synthesize Problem Space Information

- Begin to gather the information specific to the domain of focus
  - Reverse engineer domain systems
  - Gather information from other Government and DoD agencies
- The analysis thus far is a reusable asset - provide to ARC

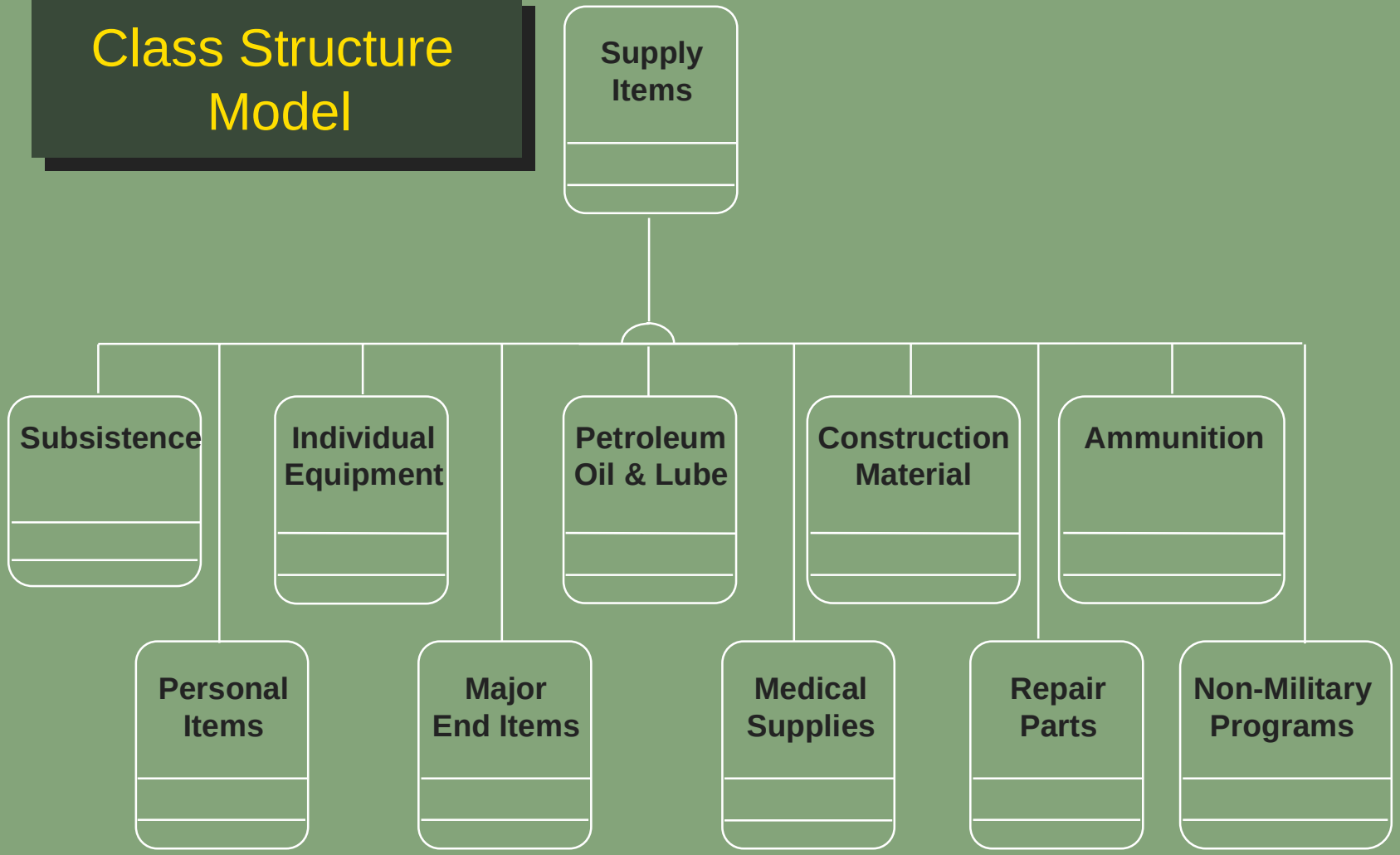
Context Model



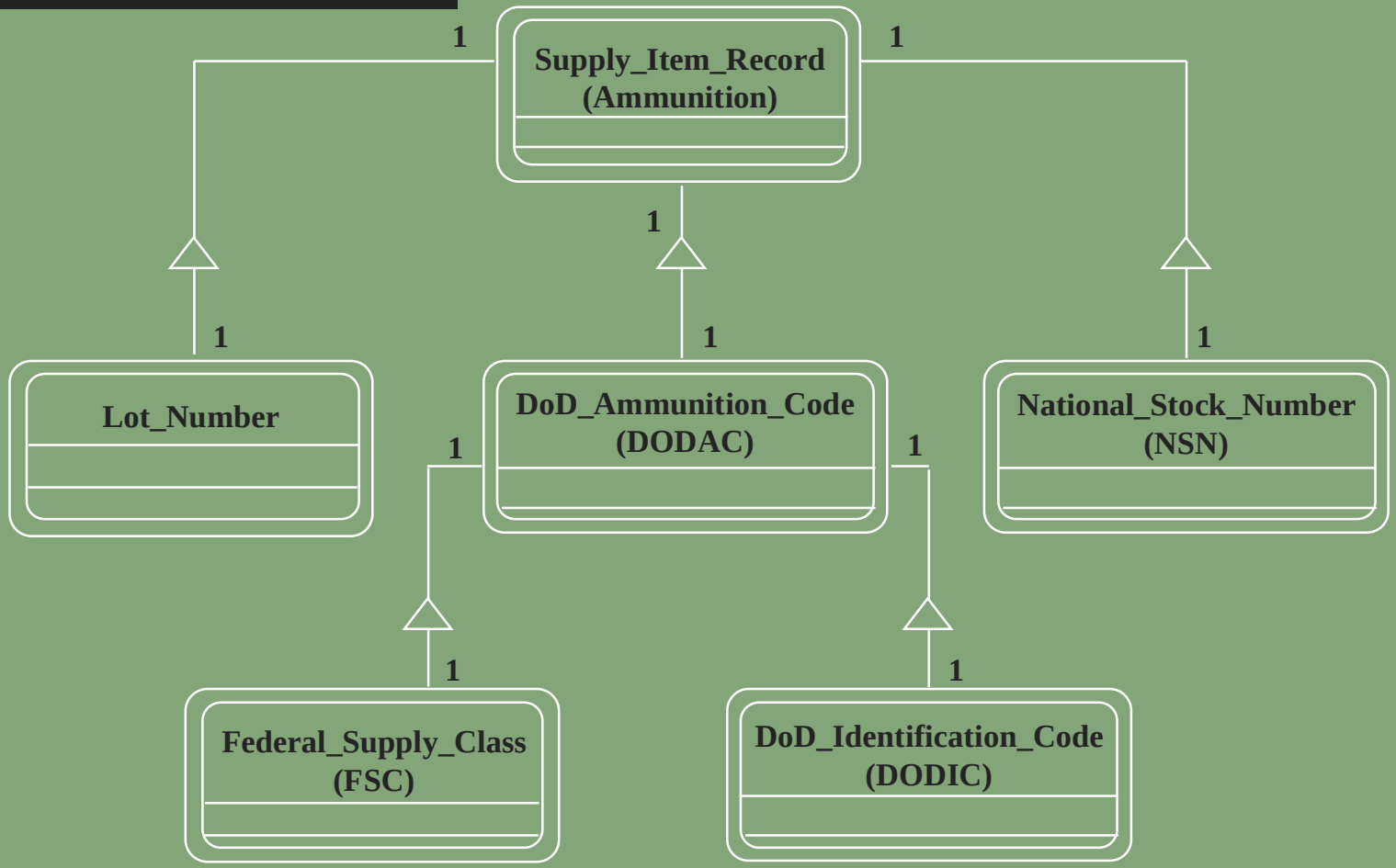
Behavioral Model (ERD)



Class Structure Model

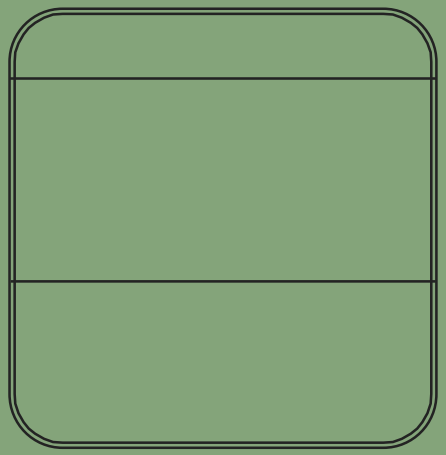


Assembly Structure Model



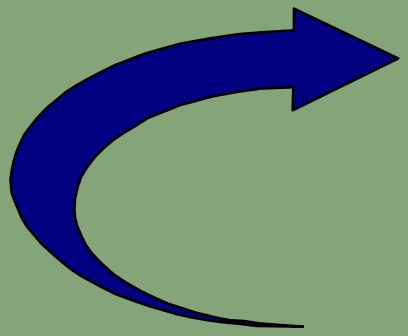
Identify Commonalties

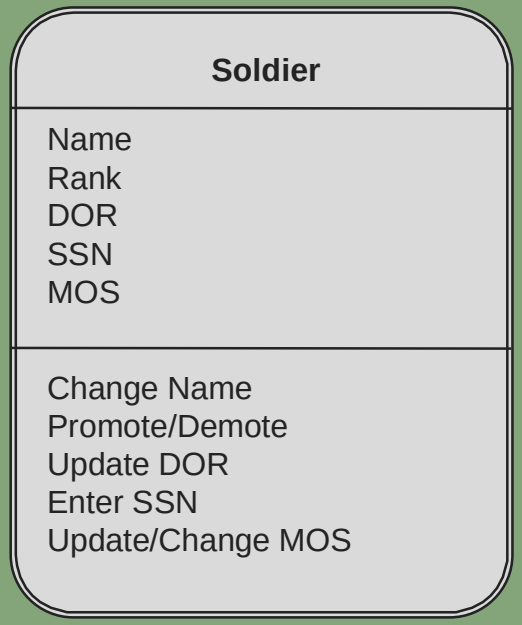
- Identify objects & their relationships
  - Determine composition
  - Identify structure
  - Determine connection
- Determine behavior
- Identify constraints
- Develop and verify common objects model



Identify Objects

An object



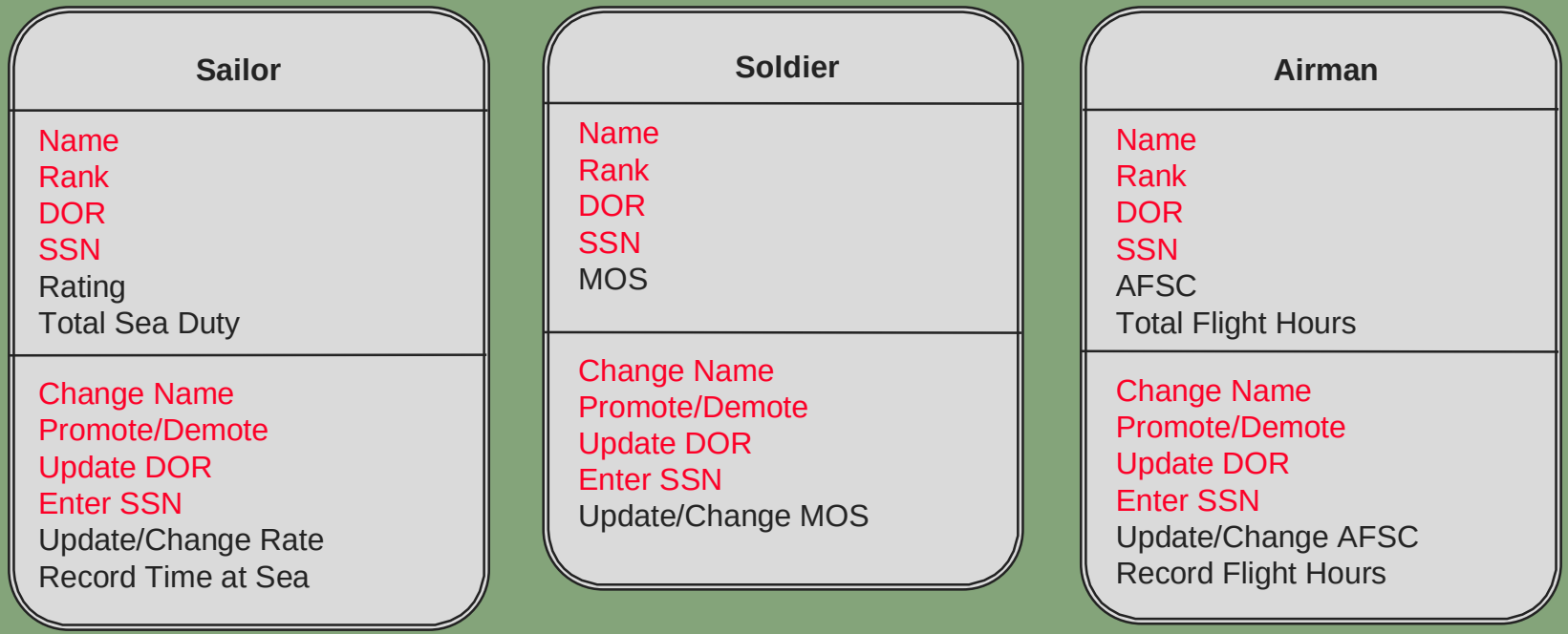


Object Name

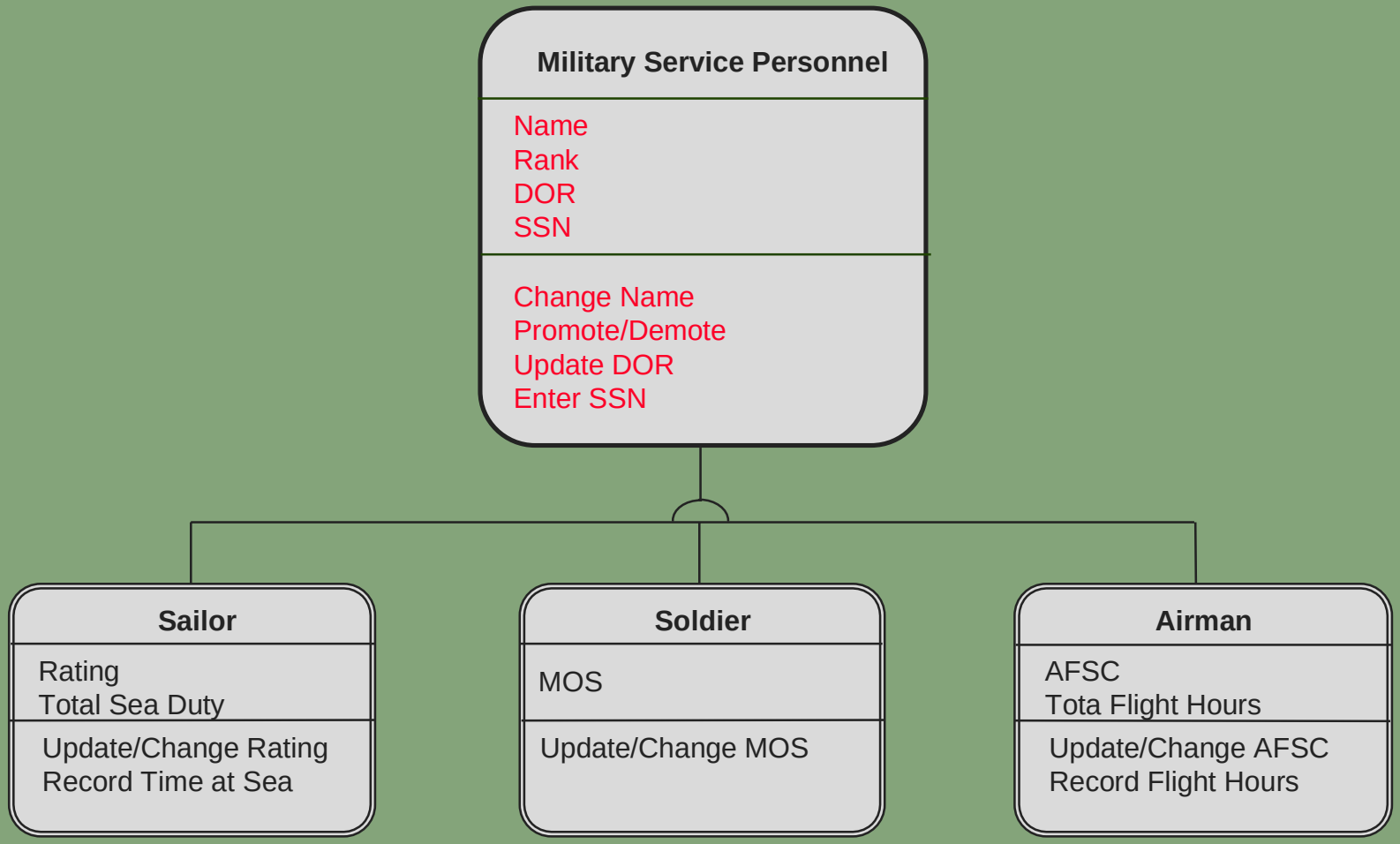
Attributes

Services

Identify Commonalities

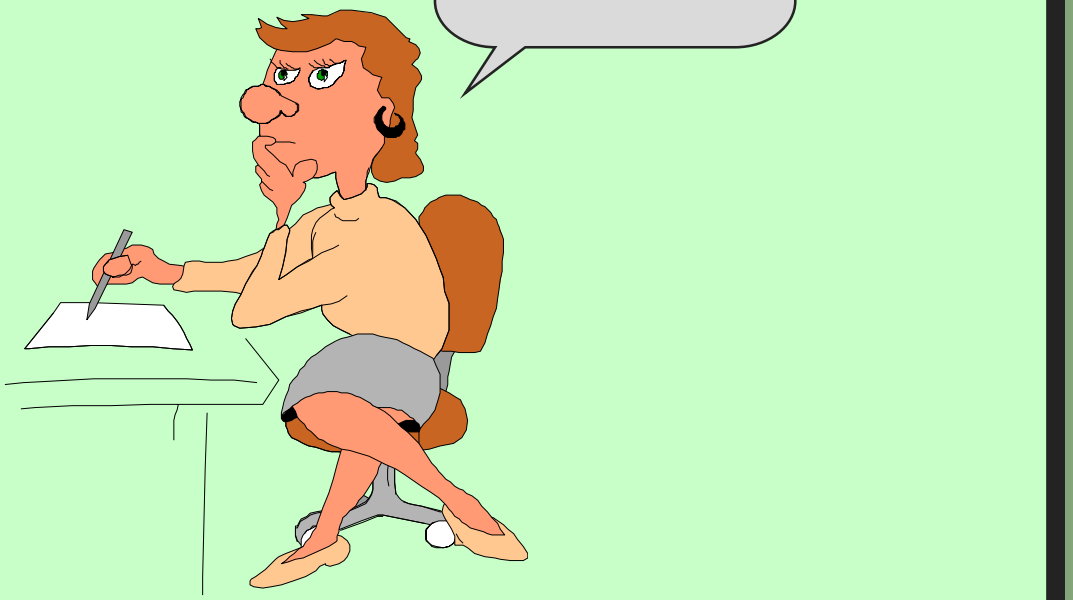


Promote Commonalties

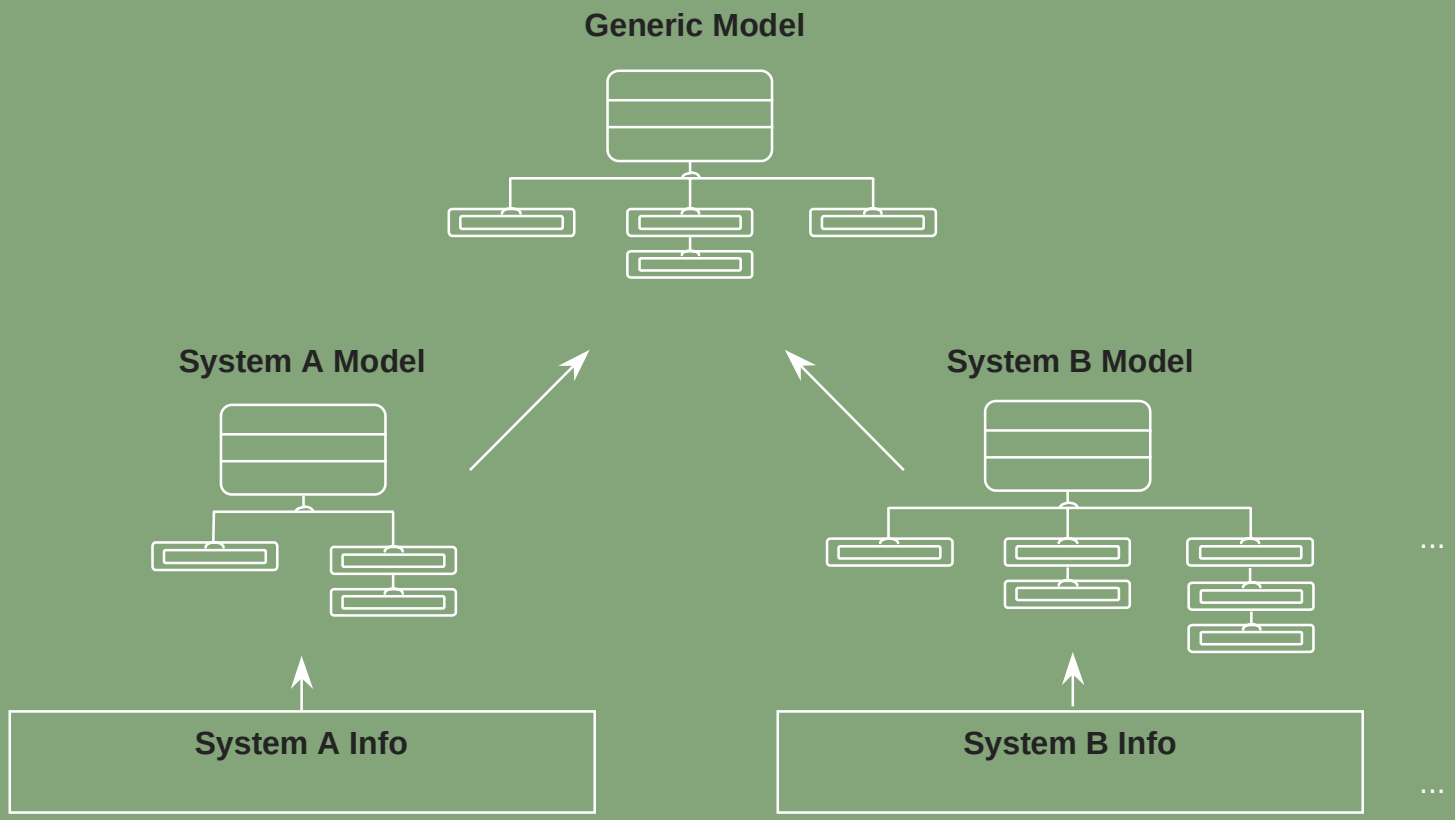


Domain Lexicon

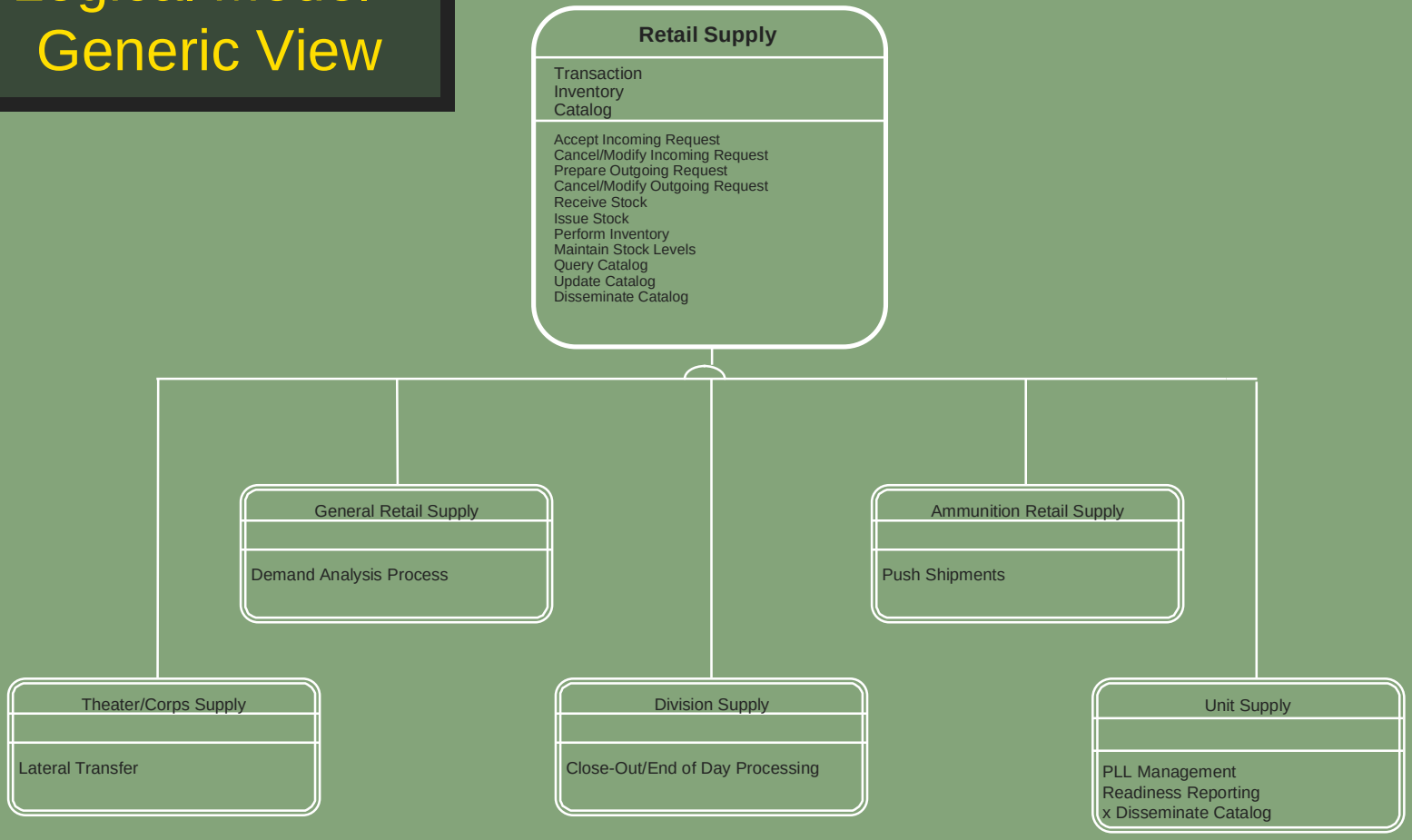
Term	Description
Authenticate	
Signature	
Sign Form	
Approve	



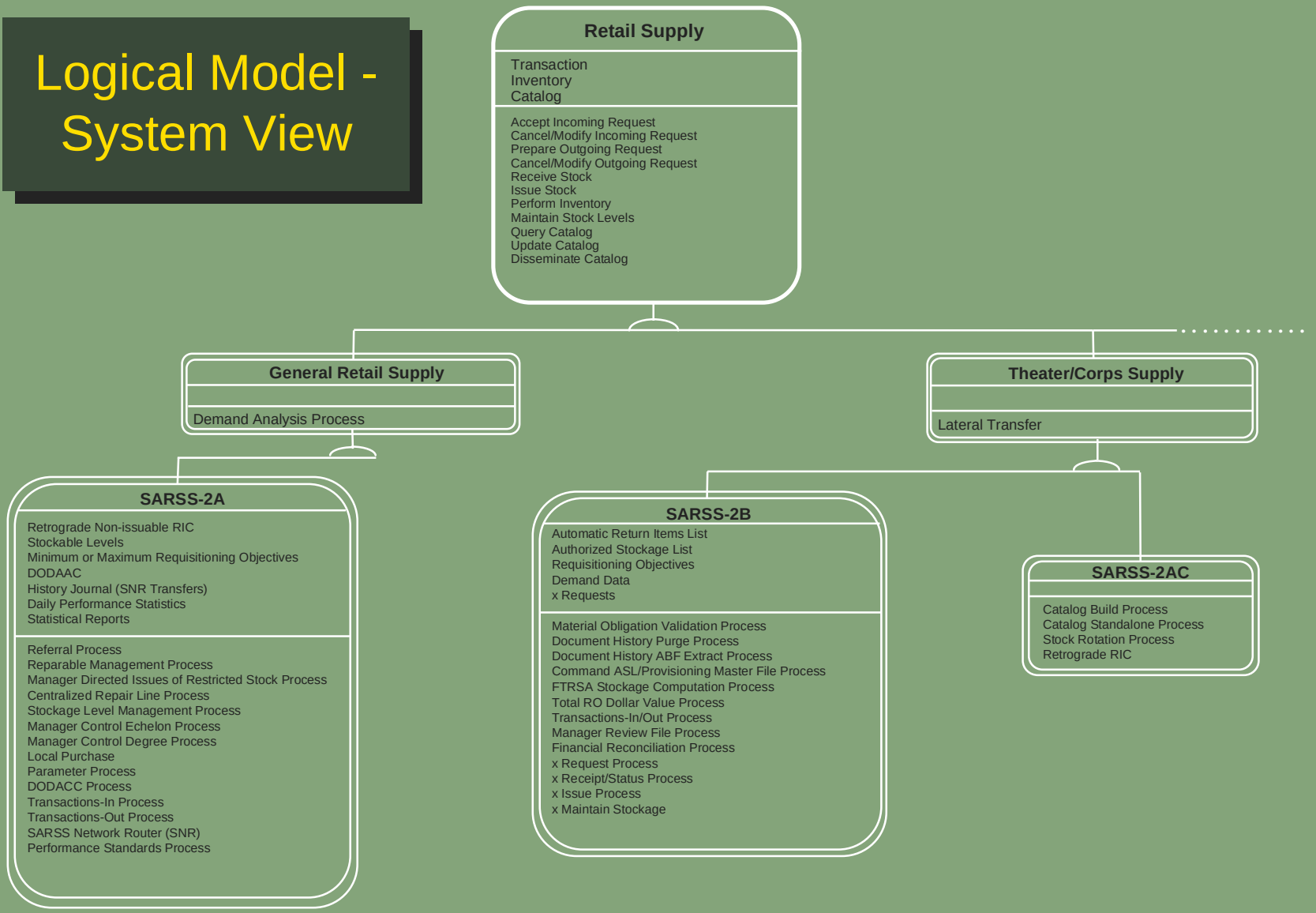
The Modeling Process



Logical Model - Generic View



Logical Model - System View



Class/Object Specifications

Class/Object Name: Retail Supply System
Description: The Retail Supply class is focused on systems that provide logistical support in the form of supply replenishment for the ten DoD classes of supply. Most systems that have direct (SAARS,SAAS) or indirect (SAMS) supply responsibilities will have the following data and capability characteristics.
Source(s): PEO-STAMIS Retail Supply domain analysis
Adaptation Requirements (Variants): None
Constraints: None
Concurrency: None
Structure:
- **Whole_To:** None
- **Part_Of:** None
- **Generalize_To:** General Retail, Theater/Corps, Division, Company, Ammunition Retail
- **Specialization_To:** None

Connection:
- **Instance:** None
- **Message:** None

Donor/Client Mapping

Issue Process [Element, Retail Supply]

Description: The Issue element category consists of assets that provide for release of supply items to a customer in response to a valid request for issue. Supplies are normally issued with receipt documents.

Rationale: All retail supply systems require the capability to issue supplies in response to a customer's request or requisition.

Mapping:

DONOR	TYPE	CLIENT
SAAS-DAO, SAAS-4, SAMS-1 (Supply), SARSS-1, SARSS-2A, SPBS-R SPBS-R-I/TDA SAAS-1/3 ULLS-G (Supply), ULLS-A(Supply), SAMS-I/TDA (Supply)	Requirements, Design, Code	SAMS-I/TDA upgrades (Supply), SARSS upgrades, ULLS-S4 (Supply), [Retail Supply Systems]
ULLS-S4 (Supply)	Requirements, Design	SAMS-I/TDA upgrades SARSS upgrades, [Retail Supply Systems]

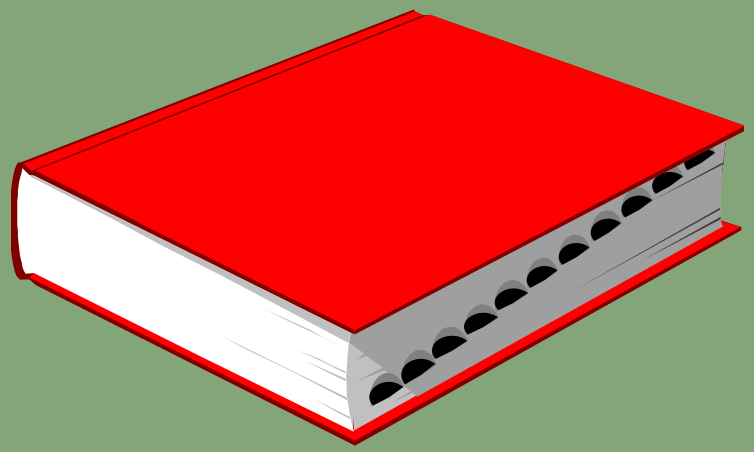
Domain Design

Identify Domain Constraints

- Management directives
- Government Off-The-Shelf (GOTS) & Commercial Off-The-Shelf (COTS)
- Hardware platform
- Common Operating Environment (COE)
  - DBMS
  - GUI
  - Communication interfaces

Collect and Organize Solution Space Information

- Develop a design catalogue
  - Existing design solutions (reusable assets)
  - Domain constraints
  - Lessons learned



Survey Domain-Specific Parts

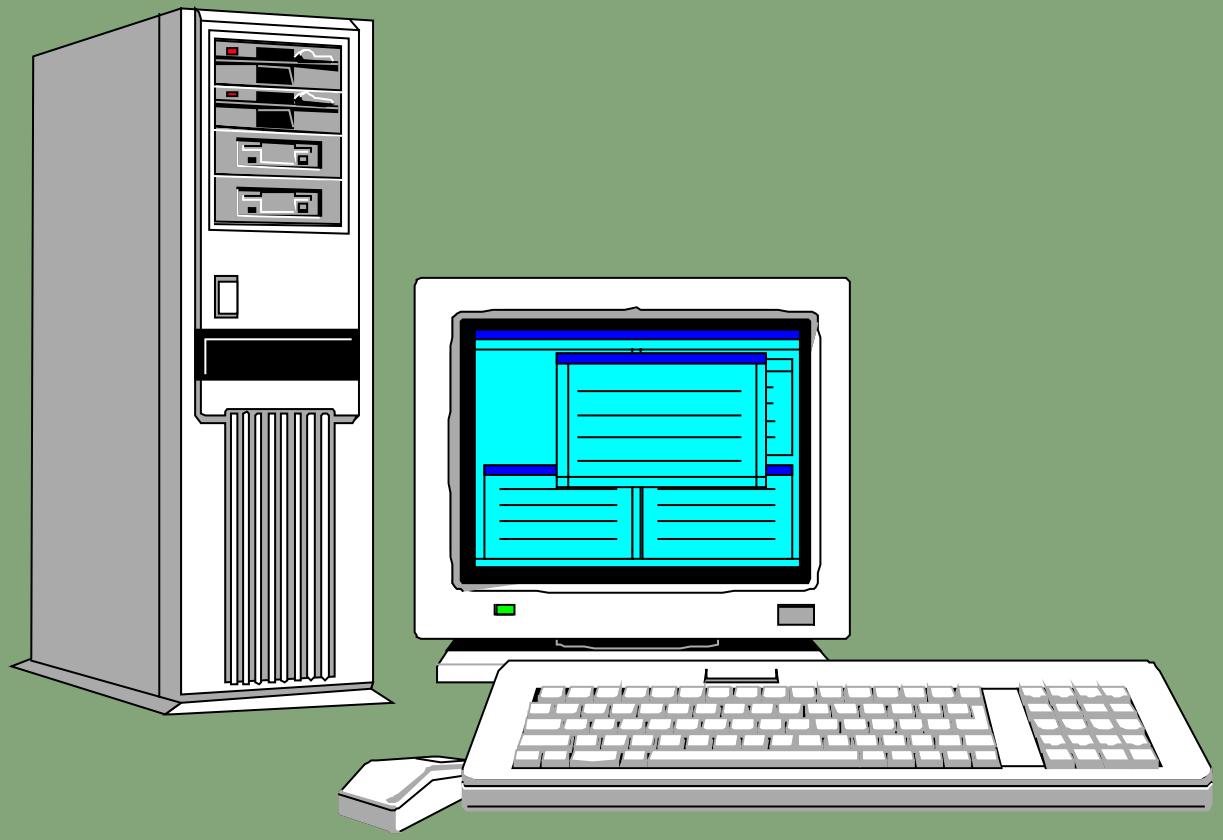
- Compare architectural components with modeled requirements
- Assess degree of reusability
- Capture classification information
- Decompose to next lower level of detail

Will it Fit?

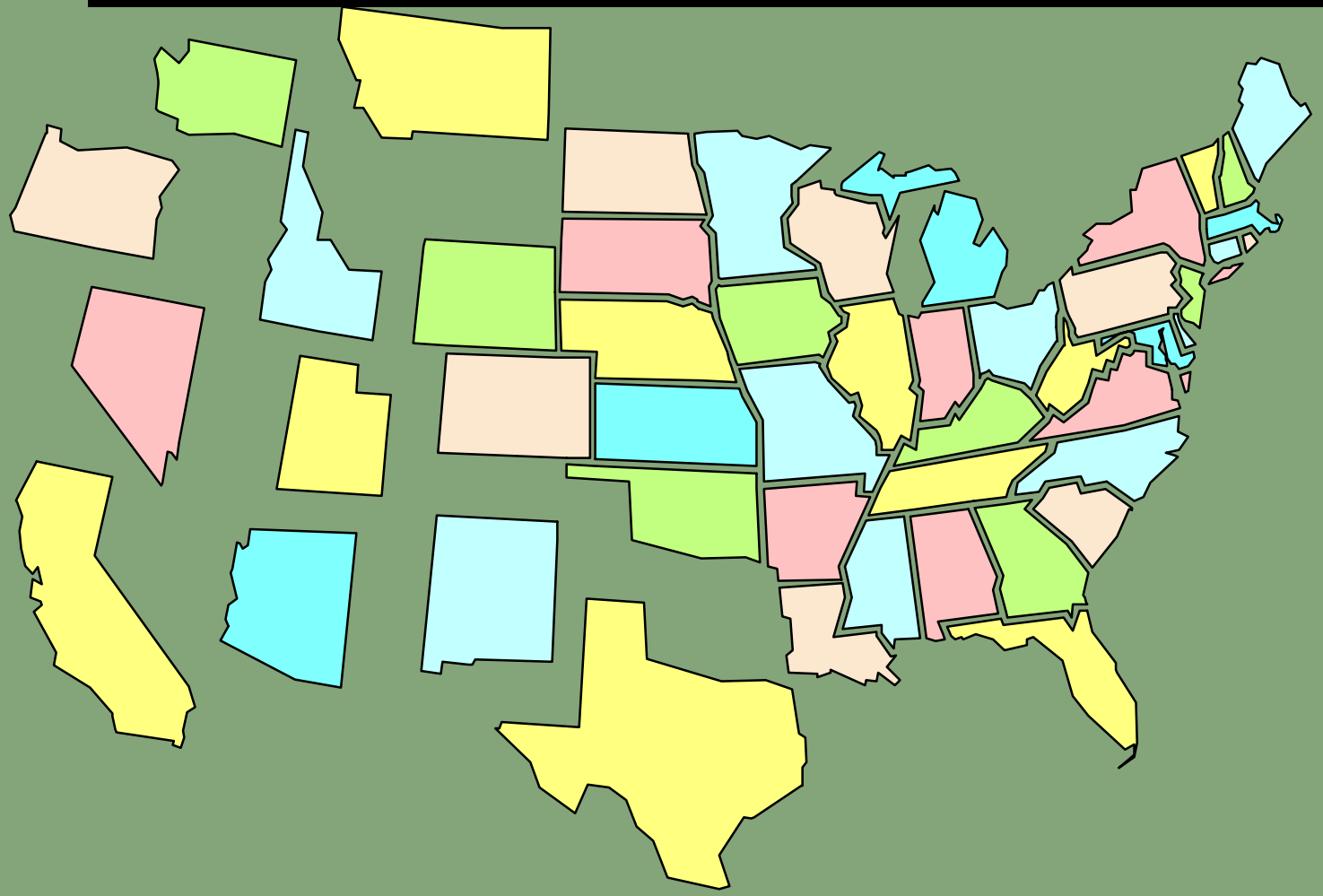

It's written in the same language.
For our operating system also.

Just how reusable is it?

Record the findings



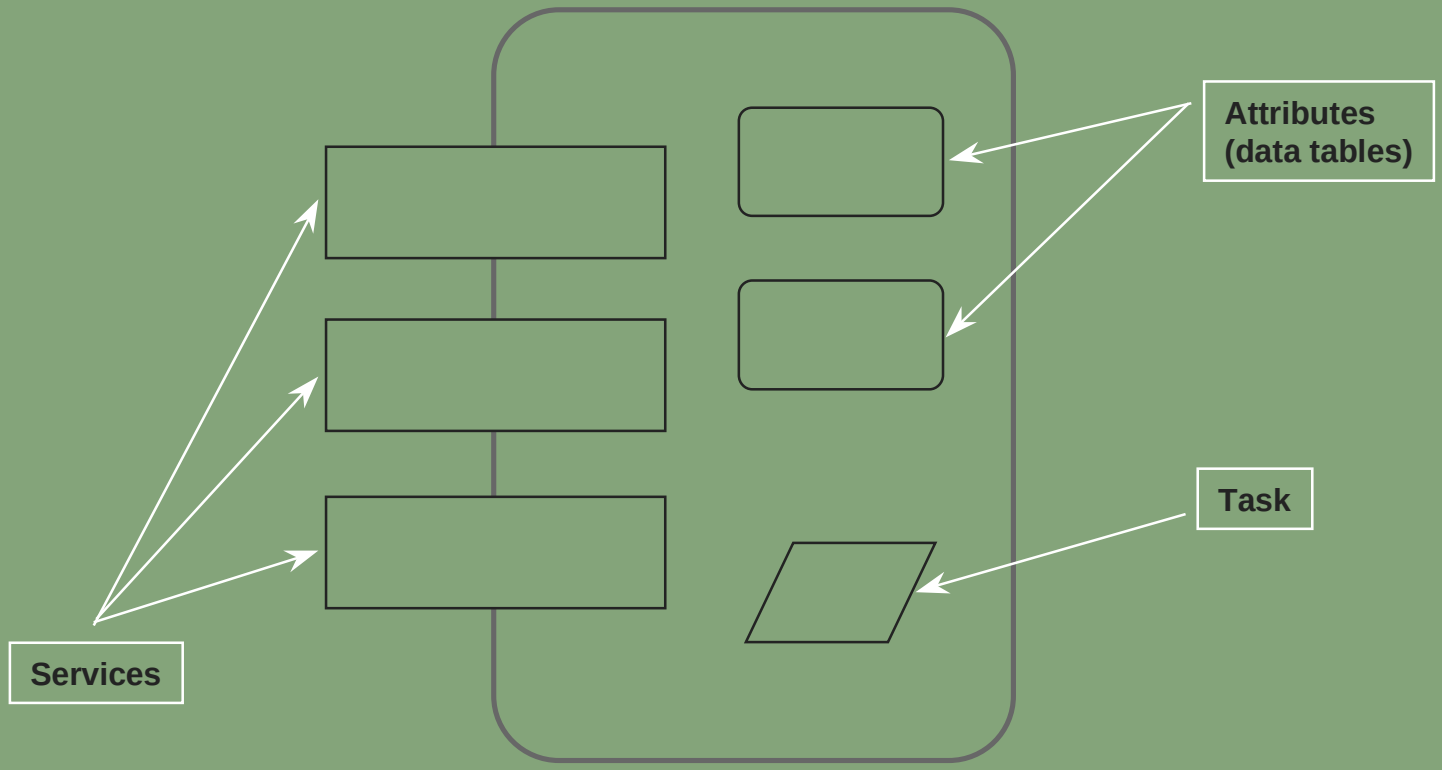
It’s only the sum of its parts



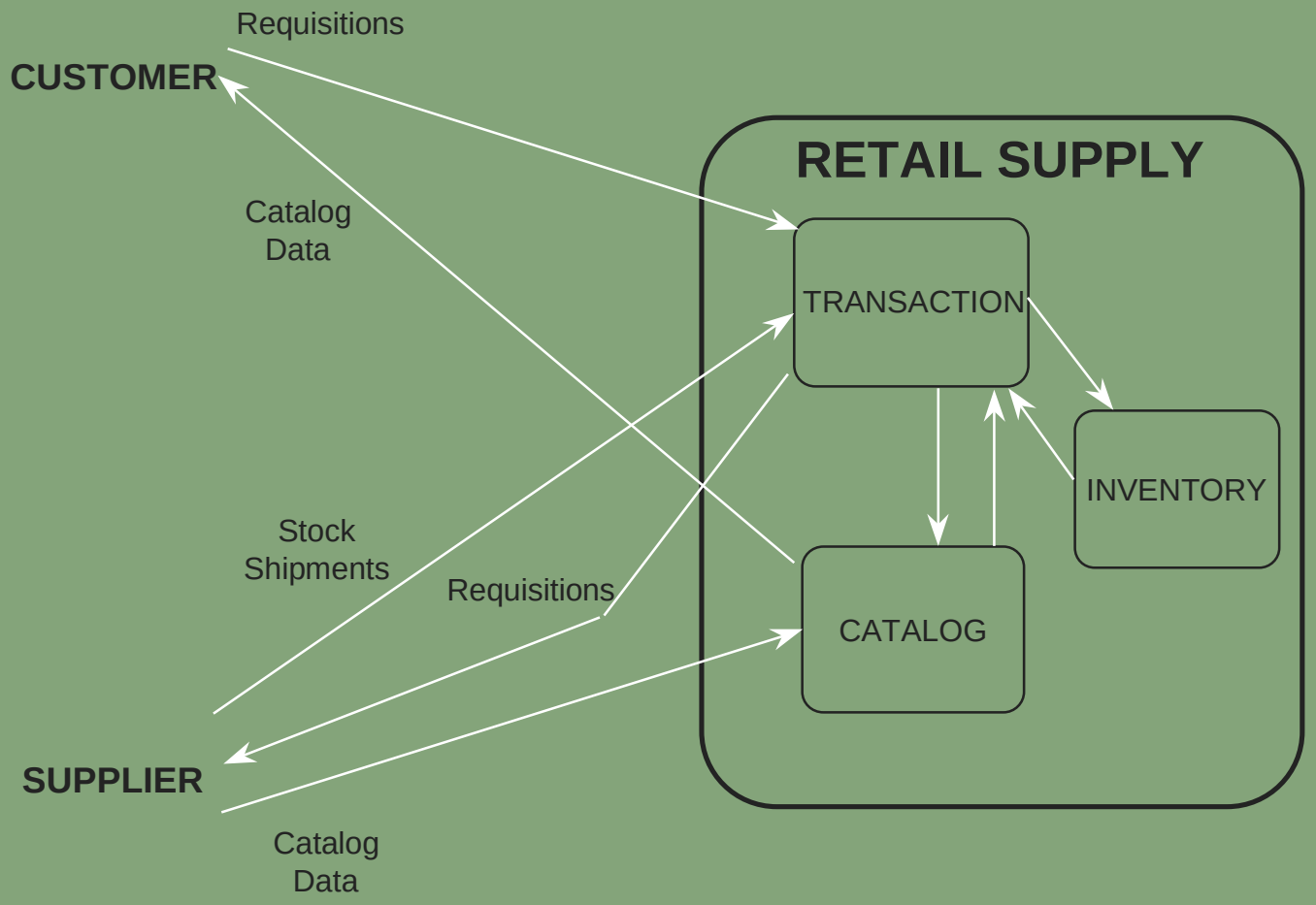
Develop Domain-Specific Software Architecture (DSSA)

- Develop high-level DSSA(s)
- Identify trade-offs and rationale for alternate DSSA(s)
- Select DSSA
- Develop specifications
- Decompose to lower-level
- Develop DSSA classification terms
- Develop DSSA guidelines

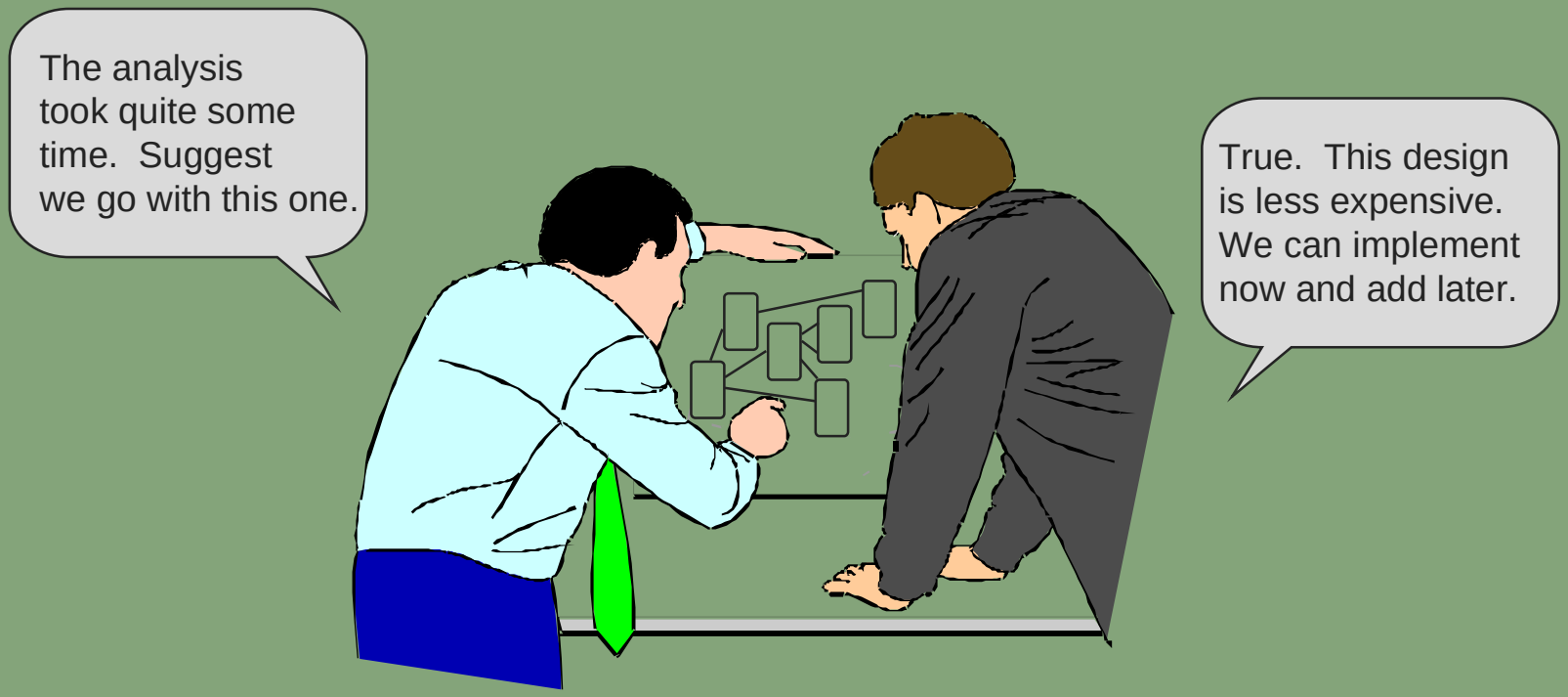
Module Diagram Icons



High-Level DSSA Diagram



Which one best meets our customers needs?

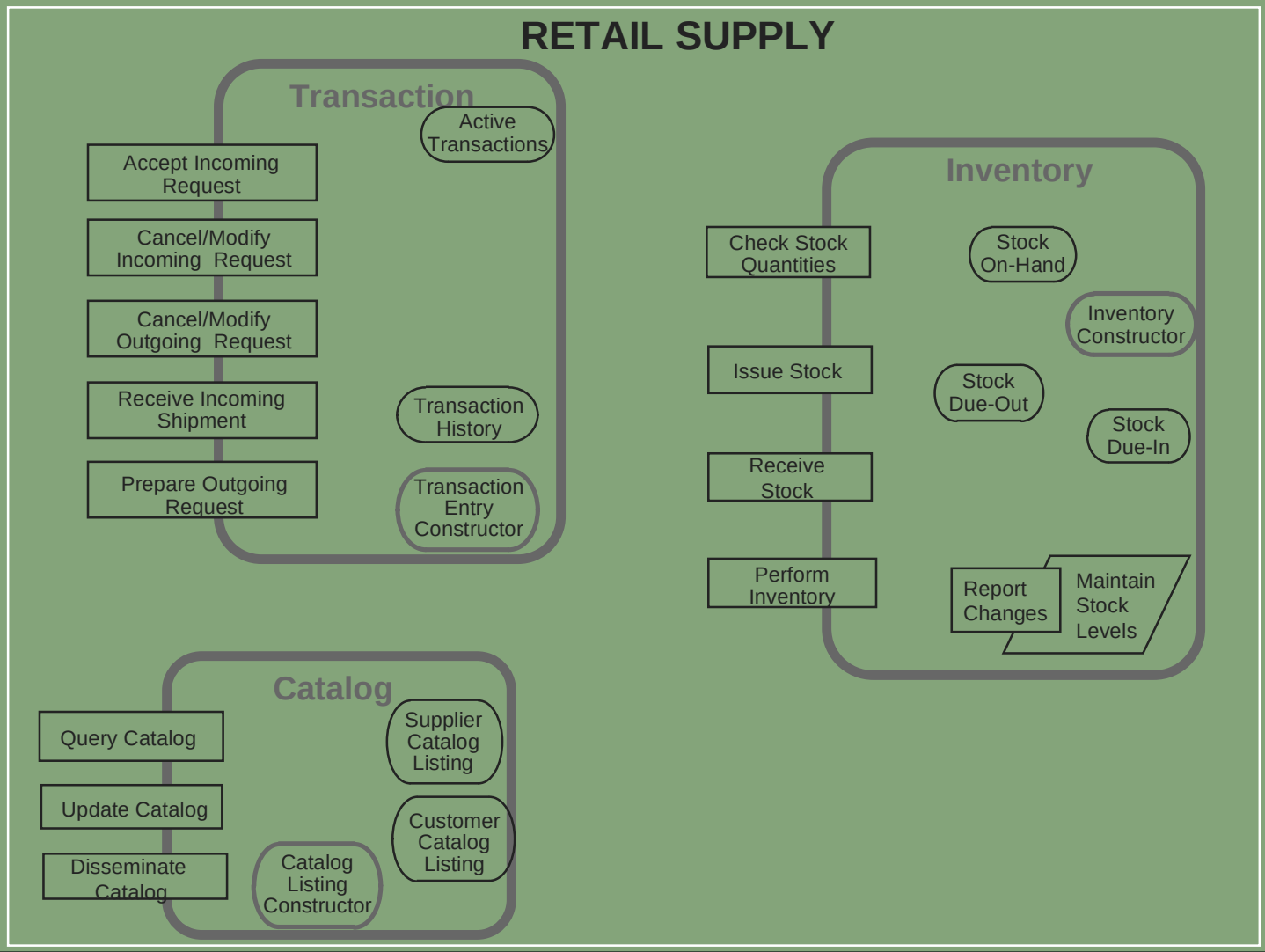


Class/Object Specifications

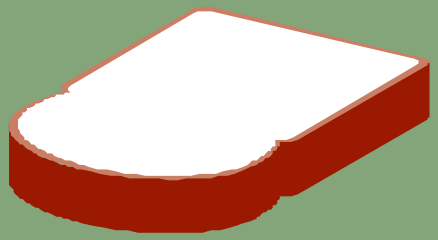
Class/Object Name: Retail Supply System
Description: The Retail Supply class is focused on systems that provide logistical support in the form of supply replenishment for the ten DoD classes of supply. Most systems that have direct (SAARS,SAAS) or indirect (SAMS) supply responsibilities will have the following data and capability characteristics.
Source(s): PEO-STAMIS Retail Supply domain analysis
Adaptation Requirements (Variants): None
Constraints: None
Concurrency: None
Structure:
- **Whole_To:** None
- **Part_Of:** None
- **Generalize_To:** General Retail, Theater/Corps, Division, Company, Ammunition Retail
- **Specialization_To:** None

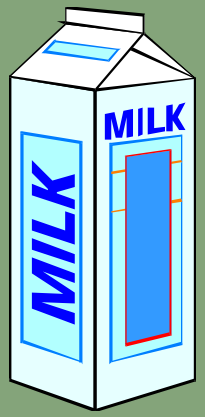
Connection:
- **Instance:** None
- **Message:** None

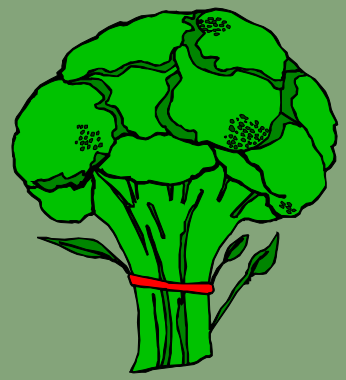
Low-Level DSSA Diagram

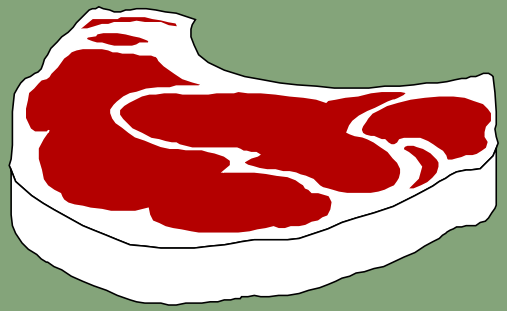


What's the best way to describe this component?

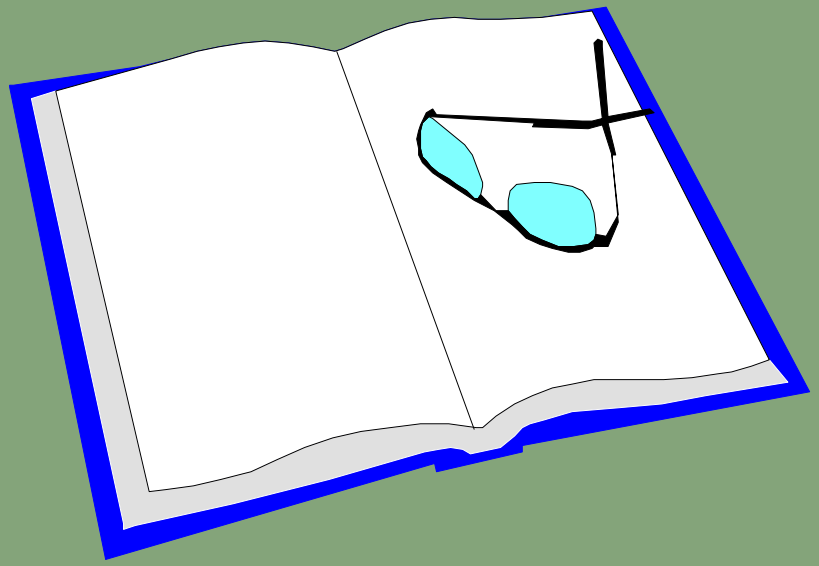








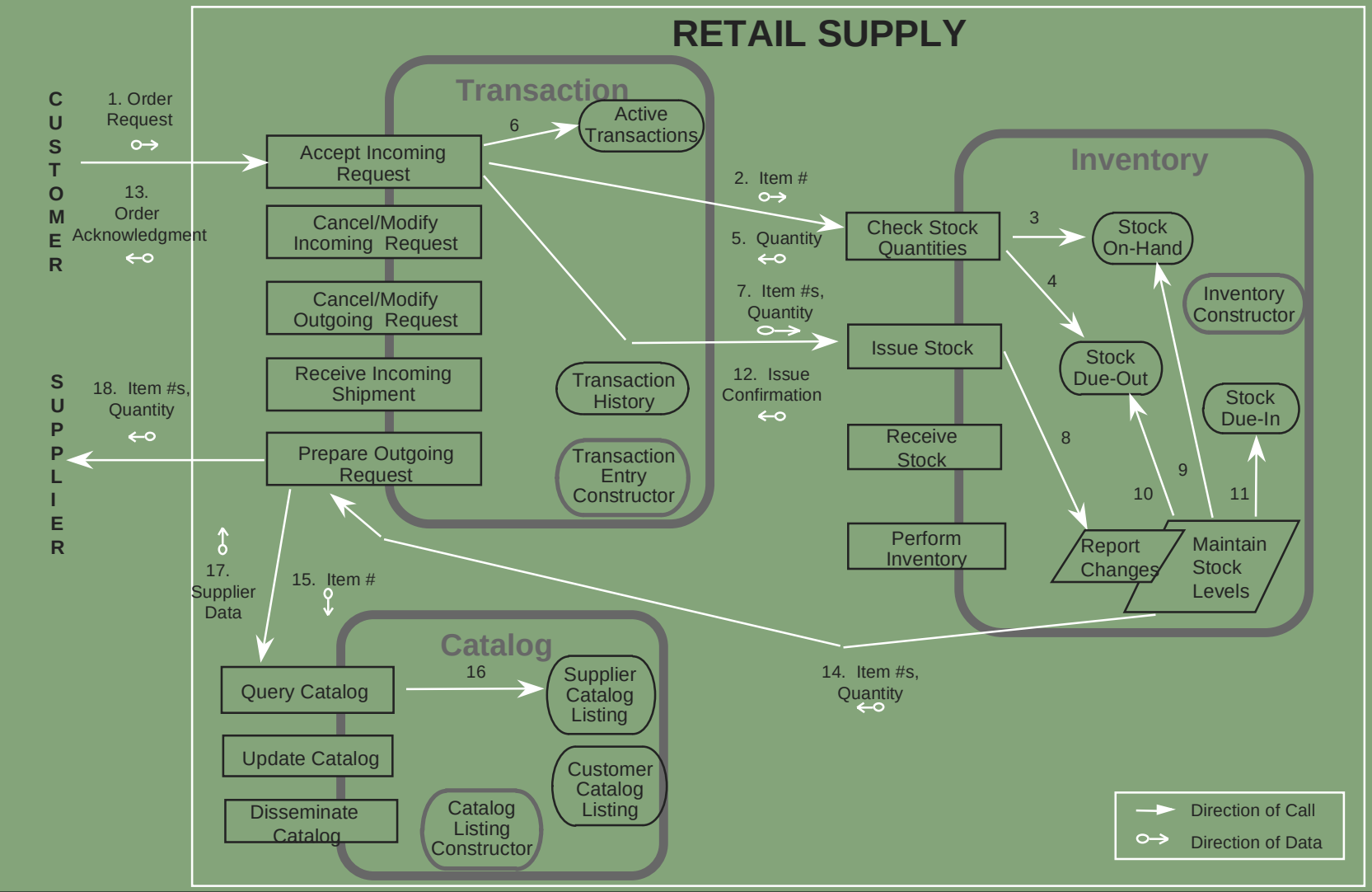
Complete and Unabridged



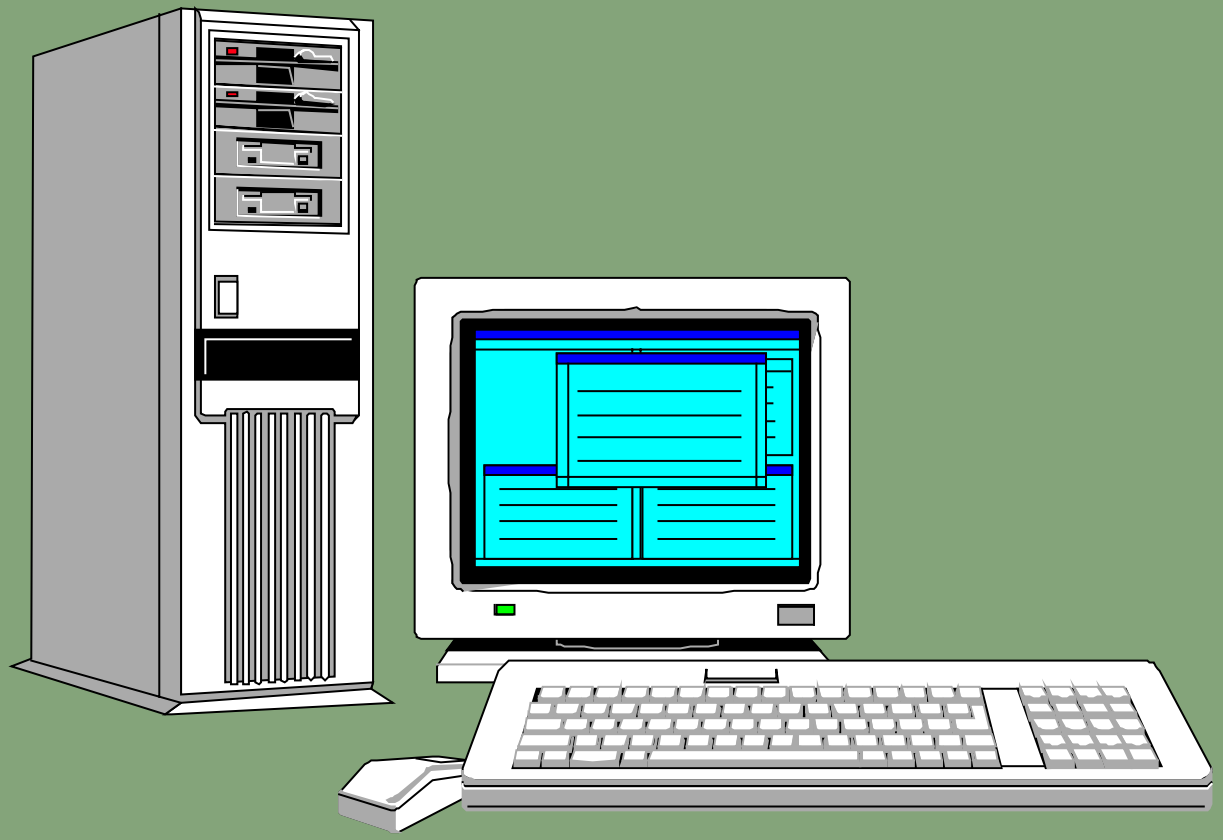
Validate DSSA

- Inspection
- Simulation
- Prototype

Validation Scenario



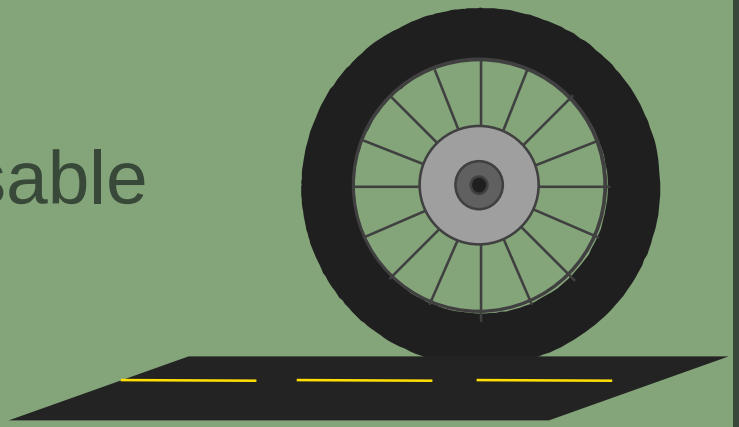
Domain Knowledge Database



Domain Implementation

The Three B's

- Borrow
  - Obtain from a reuse repository
- Buy
  - Purchasing COTS products
- Build
  - Developing a new, reusable asset



Summary

Lessons Learned

- Requires expertise in
  - Domain engineering
  - The domain/functional area
- Critical to separate problem space from solution space (what vs. how)
- Developing a domain knowledge-base at the start eliminates problems and facilitates sharing of information

Lessons Learned

- Domain Engineering is a state-of-the-art technology
- Domain Engineering products continue to evolve
- Tools are required for the Domain Engineering process

Lessons Learned

- Information sources should be kept for future Domain Engineering activities
- Availability and cooperation of domain experts is critical to the success of the project
  - Involve domain experts early to minimize errors and expedite validation process
  - Use same domain experts throughout the Domain Engineering process

Lessons Learned

- Documentation is frequently outdated
- Management's commitment and support is required
- The object-oriented paradigm facilitates identification of commonalties and their variations
- Develop a domain knowledge base/catalogue of the relevant information

Lessons Learned

- Use abstraction and partitioning to manage the complexity of the domain